OSCILLATORY NEUROCOMPUTERS WITH DYNAMIC CONNECTIVITY

The present application claims priority rights based on U.S. Provisional Application Serial No. 60/108,353 filed November 13, 1998.

FIELD OF THE INVENTION

The present invention relates generally to computational devices and more particularly to a neural network computer requiring a minimal number of connective devices between processing elements.

BACKGROUND OF THE INVENTION

Artificial neural networks, or neurocomputers, are biologically inspired; that is, they are composed of elements that perform in a manner analogous to the most elementary functions of the biological neuron. Typically, a neurocomputer is composed of a number (n) of processing elements that may be switches or nonlinear amplifiers. These elements are then organized in a way that may be related to the anatomy of the brain. The configuration of connections, and thus communication routes, between these elements represents the manner in which the neurocomputer will function, analogous to that of a program performed by digital computers. Despite this superficial resemblance, artificial neural networks exhibit a surprising number of the brain's characteristics. For example, they learn from experience, generalize from previous examples to new ones, and abstract essential characteristics from inputs containing irrelevant data. Unlike a von Neumann computer, the neurocomputer does not execute a list of commands (a

program). Rather, the neurocomputer performs pattern recognition and associative recall via self-organization of connections between elements.

Artificial neural networks can modify their behavior in response to their environment. Shown a set of inputs (perhaps with desired outputs), they self-adjust to produce consistent responses. A network is trained so that application of a set of inputs produces the desired (or at least consistent) set of outputs. Each such input (or output) set is referred to as a vector. Training is accomplished by sequentially applying input vectors, while adjusting network weights according to a predetermined procedure. During training, the network weights gradually converge to values such that each input vector produces the desired output vector.

Because of their ability to simulate the apparently oscillatory nature of brain neurons, oscillatory neurocomputers are among the more promising types of neurocomputers. Simply stated, the elements of an oscillatory neurocomputer consist of oscillators rather than amplifiers or switches. Oscillators are mechanical, chemical or electronic devices that are described by an oscillatory signal (periodic, quasi-periodic, almost periodic function, etc.). Usually the output is a scalar function of the form $V(\omega t+\phi)$ where V is a fixed wave form (sinusoid, saw-tooth or square wave), ω is the frequency of oscillation, and ϕ is the phase deviation (lag or lead).

Recurrent neural networks have feedback paths from their outputs back to their inputs. As such, the response of such networks is dynamic in that after applying a new input, the output is calculated and fed back to modify the input. The output is then recalculated, and the process is repeated again and again. Ideally, successive iterations produce smaller and smaller output changes until eventually the outputs become constant. To properly exhibit associative and recognition properties, neural networks,

such as is required by Hopfield's network, must have a fully connected synaptic matrix. That is, to function optimally, recurrent network processing elements must communicate data to each other. Although some prototypes have been built, the commercial manufacture of such neurocomputers faces a major problem: A conventional recurrent neurocomputer consisting of n processing elements requires n^2 connective junctions to be fully effective. The terms connector or connective junction, as used herein throughout, are defined as a connective element that enables one processing element to receive as input data output data produced by itself or any other one processing element. For large n this is difficult and expensive.

Accordingly, a need exists for a neurocomputer with fully recurrent capabilities and requiring a minimal number of connective devices between processing elements.

SUMMARY OF THE INVENTION

In accordance with the present invention, a neurocomputer is disclosed that exhibits pattern recognition and associative recall capabilities while requiring only n connective junctions for every n processing elements employed thereby.

In a preferred embodiment of the invention, the neurocomputer comprises n oscillating processing elements that can communicate through a common medium so that there are required only n connective junctions. A rhythmic external forcing input modulates the oscillatory frequency of the medium which, in turn, is imparted to the n oscillators. Any two oscillators oscillating at different frequencies may communicate provided that the input's power spectrum includes the frequency equal to the difference between the frequencies of the two oscillators in question. Thus, selective communication, or dynamic connectivity, between different neurocomputer oscillators occurs due to frequency modulation of the medium by external forcing.

BRIEF DESCRIPTION OF THE DRAWING

FIG. 1 is a schematic diagram of a prior art recurrent neural network employing five neural processing elements.

FIG. 2 is a schematic diagram of a neural network according to principles of the present invention and employing five neural processing elements.

FIG. 3 is a diagrammatic illustration of results obtained through simulation of the neurocomputer according to principles of the present invention using a phase deviation model and Hebbian learning rule with parameters: $n = 60$, $t \in [0, 10]$.

FIG. 4 is a schematic block diagram of a phase-locked loop.

FIG. 5 is a diagrammatic illustration of the relationship between demodulated output voltage and input frequency and phase of a phase-locked loop as depicted in FIG. 4.

FIG. 6 is a schematic block diagram of a neural network according to principles of the present invention employing two phase-locked loops as depicted in FIG. 4.

FIG. 7 is a diagrammatic illustration of one frequency multiplication performed in the neural network depicted in FIG. 6.

FIG. 8 is a schematic diagram showing the circuit components of the neural network of FIG. 6 according to principles of the present invention.

FIG. 9 is a schematic block diagram of a five-oscillator neural network with associated function generator and oscilloscopes connected for testing.

FIGS. 10A-10E are oscilloscope traces of the oscillator responses of the network of FIG. 9 with a signal sin t impressed on the input.

FIGS. 11A-11E are oscilloscope traces of the oscillator responses of the network of FIG. 9 with a signal sin 2t impressed on the input.

FIGS. 12A-12E are oscilloscope traces of the oscillator responses of the network of FIG. 9 with a signal sin 3t impressed on the input.

FIGS. 13A-13E are oscilloscope traces of the oscillator responses of the network of FIG. 9 with a signal sin 10t impressed on the input.

DESCRIPTION OF THE PREFERRED EMBODIMENT

FIG. 1 schematically illustrates a conventional recurrent neurocomputer 10 comprising n (in this case, n=5) neural processing elements 20. Elements 20 may comprise switches, amplifiers, oscillators or any other suitable neurocomputer element type known in the art. In order for each of elements 20 to communicate with the others of elements 20, neurocomputer 10 necessarily includes n^2 (in this case, n^2 =25) connective junctions 30 to which conductors 40 are attached. As can be observed, where the number n of elements 20 grows large, the implementation of such a neurocomputer becomes prohibitively difficult, from both cost and practicability standpoints.

FIG. 2 schematically illustrates a neurocomputer 50 according to principles of the present invention. ~~Neurocomputer 50 comprises a finite number n (in this case, n=5) oscillatory neural processing elements 60A, 60B, 60C, 60D and 60E.~~ Elements 60A, 60B, 60C, 60D and 60E can comprise voltage-controlled oscillators, optical oscillators, lasers, microelectromechanical systems, Josephson junctions, macromolecules, or any other suitable oscillator known in the art. Each element 60A, 60B, 60C, 60D and 60E oscillates at a particular frequency that may or may not be the same frequency as that of

the others of elements 60A, 60B, 60C, 60D and 60E. In its most general sense, the neurocomputer 50 further comprises a medium 70 connected to each of elements 60A, 60B, 60C, 60D and 60E by means of connective junctions 80A, 80B, 80C, 80D and 80E, respectively. Medium 70 may comprise a unitary body or multiple connected bodies. Neurocomputer 50 further comprises a rhythmic forcing signal source 90 able to apply a modulated oscillatory frequency to medium 70 by means of a connection 100. Specifically, the medium 70 can be a conductive medium electrically connected to the oscillators 60A, 60B, 60C, 60D and 60E by conductive connection junctions 80A, 80B, 80C, 80D and 80E. The rhythmic forcing signal source 90 can be an electrical signal generator such as a frequency modulated transmitter connected by a conductive connection 100 to the medium 70.

In operation, any two elements, such as 60B and 60E, can be said to communicate to each other if changing the phase deviation of one influences the phase deviation of the other. Such is the case if the two elements oscillate at the same frequency. Accordingly, if elements 60B and 60E oscillate at the same frequency, they will communicate in such manner.

If elements 60B and 60E oscillate at different frequencies, they will not communicate in such manner. However, by causing input signal source 90 to apply a uniform oscillatory signal multiplicatively to elements 60A, 60B, 60C, 60D and 60E by way of medium 70, any two oscillators, such as 60B and 60E, can be made to communicate by filling the frequency gap between them. That is, the uniform oscillatory signal must include a frequency equal to the difference between the respective frequencies of elements 60B and 60E. Accordingly, if elements 60B and 60E are oscillating at two different frequencies, say $\omega 1$ and $\omega 2$, then applying the time (t)

dependent voltage signal $a(t)=\cos(\omega 1-\omega 2)t$ to medium 70 enables elements 60B and 60E to communicate data to each other.

Mathematical analysis of the said neurocomputer architecture, which is based on the theory developed by F. C. Hoppensteadt and E. M. Izhikevich (Oscillatory neurocomputers with dynamic connectivity, Physical Review Letters 82(1999)2983-2986) shows that the neurocomputer dynamic is equivalent to a fully connected Hopfield network (J.J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, Proceedings of National Academy of Sciences (USA) 79(1982)2554-2558). In particular, we use the well-known Hebbian learning rule (D.O.Hebb, The Organization of Behavior, J. Wiley, New York, 1949; and S. Grossberg, Non-linear neural networks: Principles, mechanisms and architectures, Neural Networks 1(1988)17-61) to show that a network of $n = 60$ oscillators can memorize and successfully retrieve through associative recall three patterns corresponding to the images "0", "1", "2", as we illustrate in Figure 3. Thus, the neurocomputer can act as a classical fully connected Hopfield network despite the fact that it has only n interconnections.

As discussed below, a neurocomputer according to principles of the present invention may be comprised mainly of phase-locked loops, amplifiers, and band-pass filters. A schematic of such a neurocomputer is shown in FIG. 8. In this discussion, emphasis will be placed on the operation of phase-locked loops, which are ideally designed to perform frequency demodulation and frequency multiplication. Because of these qualities, they are highly suited for simulating neuron interaction.

A schematic of the major components of a phase locked loop ("PLL") 110 is shown in FIG. 4. The major components include a phase detector 120, low-pass filter 130, unity amplifier 140, and a voltage controlled oscillator ("VCO") 150. Phase locked loops use a feedback loop to produce a replica of an input signal's frequency. They are similar to operational amplifiers ("op-amps") in that an op-amp amplifies the voltage difference between input signals. The difference is that a PLL amplifies the frequency difference of the inputs and sets them equal to each other, so that the internally generated signal in the VCO 150 is an exact replica of the input signal (pin 4 of the PLL). Once this has occurred the PLL 110 is said to be in the "locked on" state. When the two signals are "locked on," any change in the input's frequency is detected by the phase detector 120 as an error signal. This error signal is applied to the internal signal, which is a replica of the input, so that it will match the input signal's frequency. The error signal is essentially the phase difference in the signal, which is the information waveform. The encoded information is extracted from pin 7 of the PLL 110. By implementing the above technique, frequency demodulation is performed using PLLs.

PLLs may be set up to perform frequency multiplication. This is accomplished by placing an open circuit between pins 3 and 4 in FIG. 4 and inputting a second source at pin 3. Since the phase detector 120 of PLL 110 is classified as type 1, it has a highly linear XOR gate and a built-in four-quadrant multiplier. The four-quadrant multiplier allows PLL 110 to perform frequency multiplication very accurately. A PLL connected in this manner produces an output that is the frequency multiplication of the two inputs.

Before simulating neuron activity using phase locked loop circuitry, one first establishes the free running frequency, the capture-range, and the lock-range of the PLL.

The free running frequency (f_o) is ideally the center frequency level of the signal that is to be demodulated. The value for the free running frequency is obtained from

$$f_o = 1.2/(4 \bullet R1 \bullet C1) \qquad (i)$$

It should be mentioned that the resistance R1 and the capacitance C1 correspond to the values of resistor R1 and capacitor C1 in FIG. 4. The capture-range (f_c) is the frequency range over which the PLL will try to lock on to an input's frequency. ~~The following formula may be used to determine the capture-range.~~

$$f_c = 1/(2 \bullet \pi) \bullet \sqrt{((2 \bullet \pi \bullet f_L)/(3.6 \bullet 1000 \bullet C2))} \qquad (ii)$$

where C2 is the capacitance of the similarly designated capacitor in FIG. 5 and f_L is the ~~lock-range.~~

~~By evaluating the formula for the capture range, one can see that the capture range is limited by low-pass filter time constant.~~ The lock-range (f_L) is the range over which the PLL will remain in the locked on state. This range is generally larger than the capture-range and can be increased by increasing Vcc of the PLL as shown in the following equation.

$$f_L = 8 \bullet f_o / V_{cc} \qquad (iii)$$

After establishing the free running, lock and capture frequencies, it should be determined if there exists a linear relationship between the input frequency and phase, and the demodulated output voltage. This linear relationship can be determined and demonstrated as shown in FIG. 5.

Communication can occur when a signal is outside the capture range if it is conditioned by another signal. This can be demonstrated by implementing the multiple

PLL circuit 170 as shown in FIG. 6. The key to designing the circuit depicted in FIG. 6 is the ability to obtain the sum and difference of two input frequencies, which can be accomplished through multiplication.

In order to implement the multiplication operation as shown by the multiplication circle 171 or 172, one should understand the following theory:

$$\cos(\omega c)\cos(\omega m) = (1/2)\bullet[\cos(\omega c - \omega m) + \cos(\omega c + \omega m)]$$

$$\Rightarrow \text{Fourier Transform} \Rightarrow$$

$$(1/4)\bullet[\delta(f + (fc - fm)) + \delta(f + (fc + fm)) + \delta(f - (fc - fm)) + \delta(f - (fc + fm))]$$

FIG. 7 shows what occurs when multiplying 8kHz and 42kHz, as at the multiplier 171 in FIG. 7. As seen in FIG. 8, these 8kHz and 42kHz components are present in the output 172 of the multiplier 171. Also present are 50kHz and 34 kHz, the sum and difference of 8 kHz and 42 kHz, respectively. Interfering harmonics are also present. Here, 8 kHz and 42 kHz were chosen to obtain an adequate separation between the harmonics and the desired frequency components.

To resolve the problem of harmonic frequencies output by the multipliers 171, 173, band pass filters were placed in the circuit. The filter was comprised of an inductor and capacitor. To isolate the single frequency desired, the following formula was used:

$$f = 1/(\sqrt{(L\bullet C\bullet 2\bullet \pi)}) \qquad (iv)$$

This formula was used to choose the desired inductor and capacitor needed. A problem resulting from the addition of the band pass filter was a voltage drop of the input signal. To compensate for this drop, amplifiers were inserted into the circuit to raise the voltage

back up to the desired magnitude. The band pass filter amplifier circuits thus added are indicated at 175, 176 of FIG. 6.

By modulating the 8 kHz carrier frequency of the function generator 178 with a 100 Hz sine wave modulation and multiplying the modulated signal with a 42 kHz carrier frequency, the PLL 181 was able to demodulate the input signal and output the 100 Hz information signal. Similarly, the PLL 182 was able to demodulate the 100 Hz information signal. Testing of the circuit depicted in FIG. 6 demonstrates that communication can still occur even if a signal is outside the capture range of a PLL if the information signal is combined with another carrier signal.

In FIG. 8, the circuit shown in block diagram form in FIG. 6 is schematically shown in greater detail. The multipliers 172 and 173 are LM 565 phase locked loops from National Semiconductor. National Semiconductor op amps LM 324 are used in the band pass filter and amplification stages 175 and 176 along with the inductor and capacitor filtering circuit elements of the values shown. The phase locked loops 181 and 182 connected as oscillators employ the LM 565 phase locked loops from National Semiconductor. VCC_1, from the PLL oscillator 181 at the upper right, is fed back to the multiplier 173 at the lower left and VCC_2 is fed back from the PLL oscillator 182 at the lower right to the multiplier 172 at the upper left.

In FIG. 9, a five-element neural network 190 is shown. Five phase locked loop oscillators 191, 192, 193, 194 and 195 are tuned at frequencies W_1, W_2, W_3, W_4, W_5, respectively, where the ratio of frequencies is $W_1:W_2:W_3:W_4:W_5 = 1:2:3:4:5$. These oscillators are forced by a common function generator 198. The function generator is connected to the oscillators via the conductors 200 and 202 serving as the conductive medium and connectors, respectively, of the neural network 190 previously discussed.

Oscilloscopes 211, 212, 213, 214, and 215 are connected to the oscillators as illustrated to demonstrate the output signals of the oscillators responsive to various inputs from the function generator 198. A summing circuit 217 is connected into a feedback loop 218 common to each of the oscillators.

FIGS. 10-13 illustrate the responses of the oscillators 191-195, respectively, to four input signals generated by the function generator 198. FIG. 10 shows the traces of the oscilloscopes 211-215 corresponding to the oscillating signals of oscillators 191-195, respectively, when the forcing voltage from the function generator 198 is sin t. None of the oscillators 191-195 are in communication and the oscillator signals are unrelated, as shown by the traces 10A-10E.

With a function generator input sin 2t, the oscillators respond with signals as shown by the traces of FIGS. 11A to 11E. Here, the oscillators 191, 193 and 195 communicate, producing the oscilloscope traces of FIGS. 11A, 11C and 11E, and the oscillators 192 and 194 communicate producing the traces of FIGS. 11B and 11D.

With an input signal sin 3t impressed by the function generator 198, the oscillators 191-195 produce in the oscilloscopes 211-215 the traces shown at FIGS. 12A-12E, respectively. Oscillators 191 and 194 communicate, producing the traces of FIGS. 12A and 12D. Oscillators 192 and 195 communicate producing the traces shown at FIGS. 12B and 12E. The oscillator 193 is not in communication with any other of the oscillators and it produces the trace shown at FIG. 12C.

In FIGS. 13A-13E, shown are traces illustrating the response of the oscillators 191-195 when a signal sin 10t is impressed by the function generator 198. None of the oscillators here are communicating.

A further, more generalized example of implementation of the present invention will now be described using a network of n voltage controlled oscillators (known as Kuramoto's phase model) and represented by:

$$\dot{\vartheta}_i = \Omega_i + \varepsilon\, a(t) \sum_{j=1} \sin(\vartheta_j - \vartheta_i)\,, \qquad (1)$$

where $\vartheta_i \in S^1$ is the phase of the ith oscillator, a(t) is the external input and $\epsilon << 1$ is the strength of connections. We require that all differences $\Omega_i - \Omega_j$ be different when $i \neq j$.

(i) Averaging

Let $\vartheta_i(t) = \Omega_i t + \varphi_i$, then

$$\dot{\varphi}_i = \varepsilon\, a(t) \sum_{j=1}^{n} \sin(\{\Omega_j - \Omega_i\}t + \varphi_j - \varphi_i)\,. \qquad (2)$$

One can average this system to obtain

$$\dot{\varphi}_i = \varepsilon H_i(\varphi_1, \ldots, \varphi_n) + o(\varepsilon)\,, \qquad (3)$$

where

$$H_i = \lim_{T\to\infty} \frac{1}{T} \int_0^T a(t) \sum_{j=1}^{n} \sin(\{\Omega_j - \Omega_i\}t + \varphi_j - \varphi_i)\, dt$$

is the average of the right-hand side of (2).

(ii) Quasiperiodic External Input

Now suppose we are given a matrix of connections C = (c_{ij}). Let

$$a(t) = a_0 + \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} \cos(\{\Omega_j - \Omega_i\}t) \qquad (4)$$

be a time dependent external input, which is a quasiperiodic function of t. Since all $\Omega_i - \Omega_j$ differences are different for all i and j, it is easy to verify that

$$H_i = \sum_{j=1}^{n} \frac{c_{ij} + c_{ji}}{2} \sin(\varphi_j - \varphi_i)$$

If we denote $s_{ij} = (c_{ij} + c_{ji})/2$, use the slow time $\tau = \epsilon t$, and disregard the small-order term $o(\epsilon)$, then we can rewrite system (2) in the form

$$\varphi_i' = \sum_{j=1}^{n} s_{ij} \sin(\varphi_j - \varphi_i)\,, \qquad (5)$$

where $' = d/d\tau$. We see that the external input of the form (4) can dynamically connect any two oscillators provided that the corresponding c_{ij} is not zero.

(iii) Chaotic External Input

In general, the external input *a(t)* can be chaotic or noisy. It can dynamically connect the *i*th and the *j*th oscillators if its Fourier transform has a non-zero entry corresponding to the frequency $\omega = \Omega_j - \Omega_i$ since the average, H_i, would depend on the phase difference $\varphi_j - \varphi_i$ in this case.

(iv) Oscillatory Associative Memory

Since the connection matrix $S = (s_{ij})$ is symmetric, the phase model (5) is a gradient system. Indeed, it can be written in the form

$$\varphi_i' = -\frac{\partial U}{\partial \varphi_i}$$

where

$$U(\varphi_1, \ldots, \varphi_n) = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} s_{ij}\cos(\varphi_j - \varphi_i)$$

is a gradient function. The vector of phase deviations $\varphi = (\varphi_1, \ldots, \varphi_n) \in \mathbb{T}^n$ always converges to an equilibrium on the *n-torus* $\in \mathbb{T}^n$ as shown in FIG. 3. System (5) has multiple attractors and Hopfield-Grossberg-like associative properties as also shown in FIG. 3. Therefore, system (1) with external forcing has oscillatory associative memory.

(v) Hebbian Learning Rule

Suppose we are given a set of *m* key vectors to be memorized

$$\xi^k = (\xi_1^k, \xi_2^k, \ldots, \xi_n^k), \quad \xi_i^k = \pm 1, \quad k = 1, \ldots, m,$$

where $\xi_i^k = \xi_j^k$ means that the *i*th and the *j*th oscillators are in-phase $(\varphi_i = \varphi_j)$, and $\xi_i^k = -\xi_j^k$ means they are anti-phase $(\varphi_i = \varphi_j + \pi)$. A Hebbian learning rule of the form

$$s_{ij} = \frac{1}{n} \sum_{k=1}^{m} \xi_i^k \xi_j^k, \qquad (6)$$

is the simplest one among many possible learning algorithms. To get (5) it suffices to apply the external input of the form (4) with $c_{ij} = s_{ij}$ for all *i* and *j*.

(vi) Initializing the Network

To use the proposed neurocomputer architecture to implement the standard Hopfield-Grossberg paradigm, as we illustrate in FIG. 3, we need a way to present an input image as an initial condition $\vartheta(0)$, and to read the output from the network. While the latter

task poses no difficulty and can be accomplished using Fourier analysis of the "mean field activity," the former task requires some ingenuity since we do not have direct access to the oscillators.

Suppose we are given a vector $\xi^0 \in \mathbb{R}^n$ to be recognized. ~~Let us apply the external input $a(t) \quad c_{ij} = \xi_i^0 \xi_j^0$ with for a certain period of time.~~ This results in the phase deviation system of the form

$$\varphi_i' = \sum_{j=1}^{n} \xi_i^0 \xi_j^0 \sin(\varphi_j - \varphi_i)$$

It is easy to check that if, $\xi_i^0 \xi_j^0 = 1$ then $\varphi_i(t) - \varphi_j(t) \to 0$. and if $\xi_i^0 \xi_j^0 = -1$ then $\varphi_i(t) - \varphi_j(t) \to \pi$ for all i and j. Thus, the network activity converges to the equilibrium having phase relations defined by the vector ξ^0, as shown in FIG. 3. When we restore the original external input *a(t)*, which induces the desired dynamic connectivity, the recognition starts from the input image ξ^0. (We added noise to the image ξ^0 shown in FIG. 3 to enhance the effect of convergence to an attractor during recognition.)

Although the invention has been described in terms of the illustrative embodiment, it will be appreciated by those skilled in the art that various changes and modifications may be made to the illustrative embodiment without departing from the spirit or scope of the invention. It is intended that the scope of the invention not be limited in any way to the illustrative embodiment shown and described but that the invention be limited only by the claims appended hereto.